CIRCUIT AND METHOD FOR DETECTING THE PHASE OF A SERVO SIGNAL

Cross-reference to Related Applications

[1] This application is related to commonly owned U.S. Patent App. Ser. Nos. _____ entitled (Atty. Docket No. 99-S-190 (1678-22-1)) "DATA-STORAGE DISK HAVING FEW OR NO SPIN-UP WEDGES AND METHOD FOR WRITING SERVO WEDGES ONTO THE DISK," _____ (Atty. Docket No. 01-S-044 (1678-22-2)) entitled "CIRCUIT AND METHOD FOR DETECTING A SERVO WEDGE ON SPIN UP OF A DATA-STORAGE DISK", " _____ (Atty. Docket No. 01-S-047 (1678-22-3)) entitled "CIRCUIT AND METHOD FOR DETECTING A SPIN-UP WEDGE AND A CORRESPONDING SERVO WEDGE ON SPIN UP OF A DATA-STORAGE DISK", _____ (Atty. Docket No. 01-S-045 (1678-39)) entitled "A DATA CODE AND METHOD FOR CODING DATA", _____ (Atty. Docket No. 01-S-046 (1678-47)) entitled "CIRCUIT AND METHOD FOR DEMODULATING A SERVO POSITION BURST", _____ (Atty. Docket No. 01-S-054 (1678-49)) entitled "SERVO CIRCUIT HAVING A SYNCHRONOUS SERVO CHANNEL AND METHOD FOR SYNCHRONOUSLY RECOVERING SERVO DATA," which were filed on the same day as the present application and which are incorporated by reference.

Technical Field of the Invention

[2] The invention is related generally to recovering data, and more particularly to a circuit and method for detecting the phase of a servo signal so that a servo circuit can compensate for a reverse-connected read head.

BACKGROUND OF THE INVENTION

[3] As computer-software applications become larger and more data intensive, disk-drive manufacturers often increase the data-storage capacities of data-storage disks by increasing the density of the stored servo and application data.

[4] To increase the accuracy of a servo circuit as it reads the denser servo data from a data-storage disk, the manufacturer often codes the servo data. For

example, as discussed below in conjunction with **FIG. 4**, the manufacturer may use a Gray code to code the servo data.

[5] Unfortunately, if the manufacturer codes the servo data stored on a data-storage disk, then a disk drive that incorporates the disk often cannot incorporate conventional techniques — such as NRZ(Non Return to Zero)-NRZI (Non Return to Zero Interleave)-NRZ conversion — to compensate for a reverse-connected read head.

[6] **FIG. 1** is a plan view of a conventional disk drive **10**, which includes a magnetic data-storage disk ***12***, a read-write head ***14***, an arm ***16***, and a voice-coil motor ***18***. The disk ***12*** is partitioned into a number — here eight — of disk sectors ***20a-20h***, and includes a number — typically in the tens or hundreds of thousands — of concentric data tracks ***22a-22n***. Readable-writable application data is stored in respective data sectors (not shown) within each track ***22***. Under the control of the disk drive's head-position circuit (not shown in **FIG. 1**), the motor ***18*** moves the arm ***16*** to center the head ***14*** over a selected track ***22***.

[7] Referring to **FIG. 2**, conventional data servo wedges ***24*** — only servo wedges ***24a - 24c*** are shown — include servo data that allows the head-position circuit (not shown in **FIG. 2**) of the disk drive ***10*** (**FIG. 1**) to accurately position the read-write head ***14*** (**FIG. 1**) during a data read or write operation. The servo wedges ***24*** are located within each track ***22*** at the beginning — the disk ***12*** spins counterclockwise in this example — of each disk sector ***20***. Each servo wedge ***24*** includes respective servo data that indentifies the location (track ***22*** and sector ***20***) of the servo wedge. Thus, the head-position circuit uses this servo data to position the head ***14*** over the track ***22*** selected to be read from or written to. The manufacturer of the disk drive ***10*** typically writes the servo wedges ***24*** onto the disk ***12*** before shipping the disk drive to a customer; neither the disk drive nor the customer alters the servo wedges ***24*** thereafter. Servo wedges like the servo wedges ***24*** are further discussed below in conjunction with **FIG. 3** and in commonly owned U.S. Patent App. Ser. No. 09/783,801, filed February 14, 2001, entitled "VITERBI DETECTOR AND METHOD FOR RECOVERING A BINARY SEQUENCE FROM A READ SIGNAL," which is incorporated by reference.

[8] **FIG. 3** is a diagram of the servo wedge ***24a*** of **FIG. 2**, the other servo wedges ***24*** being similar. Write splices ***30a*** and ***30b*** respectively separate the servo wedge ***24a*** from adjacent data sectors (not shown). An optional servo address mark (SAM) ***32*** indicates to the head-position circuit (not shown in **FIG. 3**) that the read-write head ***14*** (**FIG. 1**) is at the beginning of the servo wedge ***24a***. A servo preamble ***34*** allows the servo circuit (not shown in **FIG. 3**) of the disk drive ***10*** (**FIG. 1**) to synchronize the sample clock to the servo signal (**FIG. 5**), and a servo synchronization mark (SSM) ***36*** identifies the beginning of a head-location identifier ***38***. The preamble ***34*** and SSM ***36*** are discussed in commonly owned U.S. Patent application numbers _____ (Atty. Docket No. 99-S-190 (1678-22-1)) entitled "DATA-STORAGE DISK HAVING FEW OR NO SPIN-UP WEDGES AND METHOD FOR WRITING SERVO WEDGES ONTO THE DISK," _____ (Atty. Docket No.01-S-044 (1678-22-2)) entitled "CIRCUIT AND METHOD FOR DETECTING A SERVO WEDGE ON SPIN UP OF A DATA-STORAGE DISK", _____ (Atty. Docket No. 01-S-047 (1678-22-3)) entitled "CIRCUIT AND METHOD FOR DETECTING A SPIN-UP WEDGE AND A CORRESPONDING SERVO WEDGE ON SPIN UP OF A DATA-STORAGE DISK", _____ (Atty. Docket No. 01-S-054 (1678-49)) entitled "SERVO CIRCUIT HAVING A SYNCHRONOUS SERVO CHANNEL AND METHOD FOR SYNCHRONOUSLY RECOVERING SERVO DATA", which are incorporated by reference. The location identifier ***38*** allows the head-position circuit to coarsely determine and adjust the position of the head ***14*** with respect to the surface of the disk ***12*** (**FIG. 1**). More specifically, the location identifier ***38*** includes a sector identifier ***40*** and a track identifier ***42***, which respectively identify the disk sector ***20*** and the data track ***22*** — here the sector ***20a*** and the track ***22a*** — that contain the servo wedge ***24a***. Because the head ***14*** may read the location identifier ***38*** even if the head is not centered over the track ***24a***, the servo wedge ***24a*** also includes head-position bursts A - N, which allow the head-position circuit to finely determine and adjust the position of the head ***14***.

[9] **FIG. 4** is a table of the Gray coded bit patterns ***50*** that form portions of the respective track identifiers ***42*** (**FIG. 3**) for sixteen adjacent tracks 0 – 15 (**FIG. 2**) and the

corresponding uncoded bit patterns ***52***. The uncoded patterns ***52*** for adjacent tracks differ by only one bit. For example, the only difference between the patterns ***52*** for the tracks 0 and 1 is that the least significant (rightmost) bit for track 0 is logic 0, and the least significant bit for track 1 is logic 1. Similarly, the Gray coded patterns ***50*** for adjacent tracks differ by only a pair of bits, or a one-bit shift in a pair of logic 1's. For example, the only difference between the patterns ***50*** for the tracks 0 and 1 is that the pair of least significant bits for track 0 are logic 0, and the pair of least significant bits for track 1 are logic 1. Moreover, the only difference between the patterns ***50*** for tracks 2 and 3 are that the pair of least significant logic 1's in the pattern ***50*** for track 2 are shifted left by one bit in the pattern ***50*** for track 3.

[10] Still referring to **FIG. 4**, the Gray coded patterns ***50*** allow the head-position circuit (not shown in **FIG. 4**) to determine the position of the read-write head ***14*** (**FIG. 1**) within ±1 track. More specifically, the tracks ***22*** (**FIG. 1**) are typically so close together that the head ***14*** often simultaneously picks up servo data from multiple tracks ***22***, particularly if the head is between two tracks ***22***. Consequently, the Gray coded patterns ***50*** are designed so that if the head ***14*** is between two tracks ***22***, it generates a servo signal (not shown in **FIG. 4**) that ideally represents the Gray coded pattern ***50*** of the closest of these two tracks, but of no other track. For example, if the head ***14*** is between tracks 2 and 3 but closer to the center of track 2 than to the center of track 3, then the servo signal ideally represents the coded pattern ***50*** in track 2. If there is noise or another disturbance on the servo signal, however, then a servo circuit (not shown in **FIG. 4**) may read the servo signal as representing track 3, hence the ±1 track accuracy in the position of the head ***14***. The head-position circuit uses this head-position information derived from the servo signal to position the head ***14*** over a desired track ***22***. Once the head-position circuit positions the head ***14*** over a desired track ***22*** such that the servo signal represents the pattern ***50*** of the desired track, the head-position circuit uses bursts A – N (**FIG. 3**) to center the head ***14*** over the desired track.

[11] Referring again to **FIG. 1**, during the manufacture of the disk drive ***10*** the head ***14*** may be reverse connected, in which case it reverses the phase of, *i.e.*, inverts,

the servo data as it reads a servo wedge ***24*** (**FIG. 2**). Although not shown, the head ***14*** typically has two leads that are connected to a servo circuit (not shown in **FIGS. 1 – 4**). The person or machine that assembles the disk-drive ***10*** may reverse the leads. If the leads are reversed, then the head ***14*** will invert the servo signal, and thus the servo data. Consequently, if left uncorrected, the inverted servo data may cause the disk drive ***10*** to malfunction. Although the manufacture can test the disk drive ***10*** and reconnect the head leads if they are reversed, such testing is often costly and time consuming.

[12] As discussed above, techniques such as NRZ-NRZI-NRZ conversion are often used to compensate for a reverse-connected read-write head ***14***. For example, the NRZ-NRZI-NRZ conversion converts data from one state to another such that the polarity of the resulting data recovered from the disk ***12*** (**FIG. 1**) is the same whether the leads of the head ***14*** are properly or reverse connected. That is, NRZ-NRZI-NRZ conversion eliminates the need to test the head connection because the recovered data has the correct polarity regardless of the polarity of the connection.

[13] Unfortunately, referring to **FIG. 4**, NRZ-NRZI-NRZ conversion cannot be used with the Gray coded patterns ***50*** because it will destroy the characteristics of the patterns ***50*** that allow the head-position circuit (not shown in **FIGS. 1- 4**) to determine the position of the read-write head ***14*** (**FIG. 1**) with ±1 track accuracy.

SUMMARY OF THE INVENTION

[14] In one aspect of the invention, a head-polarity detector includes a circuit for recovering servo data and a polarity determinator. The circuit recovers the servo data from a servo signal generated by a read-write head that is coupled to the circuit with a coupling polarity. The determinator determines the coupling polarity from the recovered servo data.

[15] Such a detector allows a servo circuit to compensate for a reversed-coupled read-write head, and thus allows a manufacturer to forego time-consuming and costly testing of the head-connection polarity.

BRIEF DESCRIPTION OF THE DRAWINGS

[16] **FIG. 1** is a plan view of a conventional disk drive that includes a magnetic data-storage disk having disk sectors and data tracks.

[17] **FIG. 2** is a magnified view of the servo wedges on the disk of **FIG. 1**.

[18] **FIG. 3** is a diagram of a servo wedge of **FIG. 2**.

[19] **FIG. 4** is a table of conventional Gray coded track identifiers and the corresponding uncoded track identifiers for adjacent tracks on the disk of **FIG. 1**.

[20] **FIG. 5** is a block diagram of a servo circuit that can determine the polarity of a read-write head connection and can compensate the servo signal if the connection is reversed according to an embodiment of the invention.

[21] **FIG. 6** is a block diagram of the synchronization-mark-and-polarity detector of **FIG. 5** according to an embodiment of the invention.

[22] **FIG. 7A** is a one-sample-at-a-time trellis diagram for a pruned, non-time-varying version of the Viterbi detector of **FIG. 6** according to an embodiment of the invention.

[23] **FIG. 7B** is a one-sample-at-a-time trellis diagram for a pruned, time-varying version of the Viterbi detector of **FIG. 6** according to an embodiment of the invention.

[24] **FIG. 7C** is a two-sample-at-a-time version of the trellis diagram of **FIG. 7B**.

[25] **FIG. 8** is a table of Gray coded track identifiers and the corresponding uncoded track identifiers for adjacent tracks on a disk according to an embodiment of the invention.

[26] **FIG. 9** is a block diagram of a disk-drive system that incorporates the servo circuit of **FIG. 5** and that may incorporate the Gray coded track identifiers of **FIG. 8** according to an embodiment of the invention.

DESCRIPTION OF THE INVENTION

[27] **FIG. 5** is block diagram of a synchronous servo circuit ***60***, which in accordance with an embodiment of the invention includes a synchronization-mark-and-polarity detector ***62*** for recovering a synchronization mark

such as the sync mark of Table I below, determining the connection polarity of a read-write head (**FIG. 9**) from the recovered sync mark, and causing a phase-compensation circuit ***64*** to adjust the phase of the servo signal if the head connection is reversed. The detector ***62*** is further discussed below in conjunction with **FIG. 6**, and in one embodiment, the circuit ***64*** includes a conventional twos-compliment inverter.

[28] The circuit ***60*** also includes a gain and filter circuit ***66***, which adjusts the gain of, filters, and equalizes the servo signal from the read-write head (**FIG. 9**). An analog-to-digital converter (ADC) ***68*** receives a sample clock (not shown) on a control bus ***70*** and generates digital samples of the servo signal from the circuit ***66***. A finite-impulse-response (FIR) filter ***72*** boosts the equalization of the samples received from the ADC ***68*** via the phase-compensation circuit ***64***, and timing and gain recovery loops ***74*** effectively synchronize the sample clock to the servo signal and maintain the gain of the circuit ***60*** at a desired level. The phase-compensation circuit ***64***, ADC ***68***, FIR ***72***, and loops ***74*** form a sample circuit ***76***. A Viterbi detector ***78*** recovers servo data, such as the location identifier ***38*** (**FIG. 3**), from the servo-signal samples generated by the loops ***74***. A decoder ***80*** decodes the recovered servo data from the Viterbi detector ***78*** in response to a Sync Mark Detect signal from the detector ***62***. A position-burst demodulator ***82*** receives samples of the servo signal from the FIR ***72*** and generates a head-position-error signal, and a processor ***84*** controls the components of the servo circuit ***60*** via the control bus ***70***. For example, the processor ***84*** causes the circuit ***64*** to invert the samples from the ADC ***68*** in response to a predetermined logic level of a Head Polarity signal from the detector ***62***. A servo-data interface ***86*** interfaces the decoder ***80***, demodulator ***82***, and processor ***84*** to a disk-drive controller (**FIG. 9**). Alternatively, as discussed below, depending on the scheme used to code the servo data, the circuit ***60*** may omit the Viterbi detector ***78*** and use the detector ***62*** to recover all of the servo data. Furthermore, although shown located between the ADC ***68*** and the FIR ***72***, the phase-compensation circuit ***64*** may be located elsewhere in the forward path of the servo circuit ***60*** such as at the input of the Viterbi detector ***78***.

[29] Still referring to **FIG. 5**, the circuit ***66***, ADC ***68***, FIR ***72***, loops ***74***, Viterbi detector ***78***, decoder ***80***, processor ***84***, and the general operation of the servo circuit ***60*** are further discussed in previously incorporated U.S. Patent App. Ser. Nos. _____ (Atty. Docket No. 99-S-190 (1678-22-1)) entitled "DATA-STORAGE DISK HAVING FEW OR NO SPIN-UP WEDGES AND METHOD FOR WRITING SERVO WEDGES ONTO THE DISK," _____ (Atty. Docket No. 01-S-044 (1678-22-2)) entitled "CIRCUIT AND METHOD FOR DETECTING A SERVO WEDGE ON SPIN UP OF A DATA-STORAGE DISK", _____ (Atty. Docket No. 01-S-047 (1678-22-3)) entitled "CIRCUIT AND METHOD FOR DETECTING A SPIN-UP WEDGE AND A CORRESPONDING SERVO WEDGE ON SPIN UP OF A DATA-STORAGE DISK",_____ _____(Atty. Docket No. 01-S-054 (1678-49)) entitled "SERVO CIRCUIT HAVING A SYNCHRONOUS SERVO CHANNEL AND METHOD FOR SYNCHRONOUSLY RECOVERING SERVO DATA". The timing-recovery loop of the loops ***74*** is further discussed in commonly owned U.S. Patent App. Ser. No. 09/387,146, filed August 31, 1999, entitled "DIGITAL TIMING RECOVERY USING BAUD RATE SAMPLING", which is incorporated by reference, and the gain-recovery loop of the loops ***74*** and the Viterbi detector ***78*** are also discussed in previously incorporated Patent App. Ser. No. 09/783,801, (Atty. Docket No. 99-S-185 (1678-21)), filed February 14, 2001, entitled "VITERBI DETECTOR AND METHOD FOR RECOVERING A BINARY SEQUENCE FROM A READ SIGNAL". The burst demodulator ***82*** is discussed in previously incorporated U.S. Patent App. Ser. No. _____ _____(Atty. Docket 01-S-045 (1678-47)) entitled "CIRCUIT AND METHOD FOR DEMODULATING A SERVO POSITION BURST".

[30] **FIG. 6** is a block diagram of the synchronization-mark-and-polarity detector ***62*** of **FIG. 5** according to an embodiment of the invention. The detector ***62*** includes a polarity-independent Viterbi detector ***100***, which recovers the sync mark from the servo signal regardless of the head-connection polarity and which includes a bank ***102*** of path-history registers PH00 – PHZ, one register for each state that the Viterbi detector ***100*** recognizes. A comparator ***104*** detects the sync mark and the head-connection polarity by comparing the recovered servo data from the Viterbi

detector ***100*** with the noninverted version of the sync mark stored in a register ***106***. The comparator ***104*** generates the Sync Mark Detect signal having one logic level when it detects the sync mark and another logic level otherwise, and generates the Head Polarity signal having one logic level when the head is properly connected to the servo circuit ***60*** (**FIG. 5**) and another logic level when the head connection is inverted. Alternatively, where the Viterbi detector ***78*** (**FIG. 5**) is omitted, the servo circuit ***60*** (**FIG. 5**) uses the Viterbi detector ***100*** to recover all of the servo data and to provide the recovered servo data to the decoder ***80***.

[31] Referring to **FIGS. 5** and **6**, the operation of the servo circuit ***60*** and the sync-mark-and-polarity detector ***62*** according to an embodiment of the invention is discussed.

[32] At the beginning of a read or write cycle, the servo circuit ***60*** synchronizes itself to the preamble of a servo wedge such as the preamble ***34*** of the servo wedge ***24a*** (**FIG. 3**). Specifically, while the read-write head (**FIG. 9**) is reading the preamble, the processor ***84*** causes the timing and gain recovery loops ***74*** to effectively synchronize the sample clock such that the ADC ***68*** samples the preamble at appropriate times. This synchronization is further discussed in commonly owned U.S. Patent App. Ser. No. 09/387,146, filed August 31, 1999, entitled "DIGITAL TIMING RECOVERY USING BAUD RATE SAMPLING", which is incorporated by reference.

[33] When the circuit ***60*** is synchronized, the processor ***84*** enables the detector ***62*** to search for and detect the sync mark and the head-connection polarity. During this search, the comparator ***104*** compares the recovered servo data from the Viterbi detector ***100*** to the stored sync mark on a bit-by-bit basis. If and when the number of the recovered servo bits that match the corresponding bits of the stored sync mark is greater than or equal to a first predetermined threshold or less than or equal to a second predetermined threshold, then the comparator transitions the Sync Mark Detect signal to an active logic level to indicate that it has detected the sync mark. Furthermore, the comparator ***104*** transitions the Head Polarity signal to one logic level if the number of matched bits is greater than or equal to the first threshold, and transitions to the Head Polarity signal to another logic level if the number of matched bits is less

than or equal to the second threshold. In one embodiment, the detector ***62*** allows the manufacturer to program the first and second predetermined thresholds to desired values. Furthermore, as discussed above in conjunction with **FIG. 6** and below in conjunction with **FIGS. 7A – 7C**, the Viterbi detector ***100*** is phase independent such that it can recover the sync mark from the servo data regardless of the connection polarity of the read-write head.

[34] More specifically, the detector ***62*** detects the sync mark and determines the head-connection polarity according to the following algorithm:

$$\text{If} \quad \sum_{i=0}^{SM\_length\ -1} SM(i) \oplus SM\_recover\ ed(i) \geq SM\_length\ - Threshold$$

Then INV = 1 (to indicate that this first comparison indicates recovery of the sync mark and that the head connection is inverted);

Else, INV=0 (to indicate that this first comparison does not indicate recovery of the sync mark and does not provide an indication of the head-connection polarity); and

$$\text{If} \quad \sum_{i=0}^{SM\_length\ -1} SM(i) \oplus SM\_recover\ ed(i) \leq Threshold$$

NINV = 1 (to indicate that this second comparison indicates recovery of the sync mark and that the head connection is not inverted);

Else, NINV=0 (to indicate that this second comparison does not indicate recovery of the sync mark and does not provide an indication of the head-connection polarity).

where SM_length equals the number of bits in the sync mark, SM equals the sync mark stored in the register ***106***, SM_recovered equals the sync mark recovered from the Viterbi detector ***100***, Threshold is the second predetermined threshold discussed above, and SM-length – Threshold is the first predetermined threshold discussed above.

[35] For example, if the SM_length = 10, SM = 0000110011, SM_recovered equals 0100110011, and Threshold = 2, then the summation of the algorithm equals the following:

(1) $0\oplus0 + 0\oplus1 + 0\oplus0 + 0\oplus0 + 1\oplus1 + 1\oplus1 + 0\oplus0 + 0\oplus0 + 1\oplus1 + 1\oplus1 = 1$

Because 1 < (Threshold = 2) < (SM-length – Threshold = 8), the comparator ***104*** sets INV = 0 and NINV = 1, which indicates that the circuit ***62*** has detected the sync mark and has determined that the head-connection polarity is not inverted. Consequently, the comparator ***104*** sets the Sync Mark Detect signal to a logic level that indicates that the sync mark is detected, and sets the Head Polarity signal to a logic level that indicates that the head connection is proper. In response to these logic levels, the processor ***84*** causes the phase compensator ***64*** to pass through the samples from the ADC ***68*** without altering the phase of the samples.

[36] But if, for example, SM_recovered = 1011001100, and the values of SM, SM-length, and Threshold are the same as above, then the summation of the algorithm equals the following:

(2) $0\oplus1 + 0\oplus0 + 0\oplus1 + 0\oplus1 + 1\oplus0 + 1\oplus0 + 0\oplus1 + 0\oplus1 + 1\oplus0 + 1\oplus0 = 9$

Because 9 > (SM_length – Threshold = 8) > (Threshold = 2), the comparator ***104*** sets INV = 1 and NINV = 0, which indicates that the circuit ***62*** has detected the sync mark and has determined that the head-connection polarity is inverted. Consequently, the comparator ***104*** sets the Sync Mark Detect signal to the logic level that indicates that

the sync mark is detected, and sets the Head Polarity signal to a logic level that indicates that the head connection is inverted. In response to these logic levels, the processor ***84*** causes the phase compensator ***64*** to invert the samples from the ADC ***68***. Alternatively, the manufacturer may disable the processor ***84*** from causing the compensator ***64*** to invert the samples, and instead swap the head leads in response to these logic levels so that the head is properly connected to the servo circuit ***60***.

[37] Alternatively, if SM_recovered = 1001001101 and the values of SM, SM-length, and Threshold are the same as above, then the summation of the algorithm equals the following:

(3) $0 \oplus 1 + 0 \oplus 0 + 0 \oplus 0 + 0 \oplus 1 + 1 \oplus 0 + 1 \oplus 0 + 0 \oplus 1 + 0 \oplus 1 + 1 \oplus 0 + 1 \oplus 1 = 7$

Because (Threshold = 2) < 7 < (SM_length – Threshold = 8), the comparator ***104*** sets INV = NINV = 0, which indicates that the circuit ***62*** has not detected the sync mark and has not determined the head-connection polarity. Consequently, the comparator ***104*** sets the Sync Mark Detect to a logic level that indicates that the sync mark has not been detected. In response to this logic level, the processor ***84*** ignores the Head Polarity signal and does not alter the setting (invert/noninvert) of the phase compensator ***64*** or instruct a technician to swap the head leads.

[38] Although in the above examples one predetermined threshold (SM_length – Threshold) equals the difference between the length of the sync mark and the other predetermined threshold (Threshold), the one threshold may have a value that is independent of the other threshold. In one embodiment, the two thresholds are set based on the levels of noise and interference expected in the servo signal.

[39] Still referring to **FIGS. 5** and **6**, because in one embodiment the Viterbi detector ***78*** recovers servo data following the sync mark — the location identifier ***38*** (**FIG. 3**) for example — before the detector ***62*** can determine the head-connection polarity, the decoder ***80*** discards the recovered servo data if the detector ***62*** determines that the head-connection polarity is reversed. This is because the detector ***78*** cannot properly recover inverted servo data. The processor ***84*** notifies the disk-drive controller

(**FIG. 9**) that the decoder ***80*** has discarded servo data, and the controller instructs the servo circuit ***60*** to restart the read or write cycle with the phase compensator ***64*** inverting the samples of the servo signal. Because restarting a read or write cycle is inefficient, the manufacturer typically programs the disk-drive controller to cause the servo circuit ***60*** to determine the head-connection polarity and set the phase-compensation circuit ***64*** during startup of the disk drive (**FIG. 9**), and to thereafter disable the circuit ***60*** from determining the head-connection polarity. For example, the disk-drive controller may cause the processor ***84*** to store the value of the Head Polarity signal during startup, set the phase-compensation circuit ***64*** appropriately based on this stored polarity value, and thereafter maintain the setting of the circuit ***64*** and ignore the Head Polarity signal.

[40] Conversely, in an embodiment where the servo data is coded such that the Viterbi detector ***100*** can recover both the sync mark and the other servo data, the polarity-detection capability of the comparator ***104*** can be omitted because the detector ***100*** is polarity independent. The servo circuit ***60***, however, may include a data inverter (not shown) between the detector ***62*** and the decoder ***80***, or at the output of the decoder ***80***, so that the recovered servo data will be in a proper form for the disk-drive controller (**FIG. 9**) if the head connection is inverted. An example of such a servo-data code is discussed below in conjunction with **FIG. 8**.

[41] **FIG. 7A** is a one-state-at-a-time trellis diagram for the Viterbi detector ***100*** of **FIG. 5** according to an embodiment of the invention where the sync mark includes pairs and only pairs of consecutive logic 1's that are separated by no fewer than two consecutive logic 0's. In one embodiment, the Viterbi detector ***100*** is a pruned, non-time-varying PR4 detector where the values to the left of the slashes are the ideal PR4 sample values, the values to the right of the slashes are the possible values of the most recent bit sampled, and k, k+1, and k+2 are the relative sample times. In one application, the sync mark has the bit pattern given in **Table I**.

Table I

Sync Mark Bit Pattern
000000001100000011000011

[42] The bit scheme of the sync mark allows the Viterbi detector ***100*** to have a reduced number of possible state transitions, *i.e.*, to be "pruned." Normally, each state S0 – S3 of the trellis diagram would have two entering branches for a total of eight branches between the states at consecutive sample times. But with the restriction on the sync-mark bit pattern described above, there can be no state transition from S1 to S2 or from S2 to S1. Therefore, eliminating these two state transitions leaves only six branches between the states at consecutive sample times.

[43] Furthermore, because the trellis of the Viterbi detector ***100*** is symmetrical about an imaginary horizontal axis ***120*** between states S1 and S2, the Viterbi detector ***100*** can recover the sync mark regardless of its polarity, and thus regardless of the head-connection polarity.

[44] The fundamentals of Viterbi detectors and trellis diagrams are further discussed in commonly owned U.S. Patent Application Ser. Nos. 09/409,923, filed September 30, 1999, entitled "PARITY-SENSITIVE VITERBI DETECTOR AND METHOD FOR RECOVERING INFORMATION FROM A READ SIGNAL", and 09/410,274, filed September 30, 1999, entitled "CIRCUIT AND METHOD FOR RECOVERING SYNCHRONIZATION INFORMATION FROM A SIGNAL", which are incorporated by reference.

[45] **FIG. 7B** is a one-state-at-a-time trellis diagram for the Viterbi detector ***100*** of **FIG. 5** according to another embodiment of the invention where the sync mark includes pairs and only pairs of consecutive logic 1's that are separated by no fewer than two consecutive logic 0's. In one embodiment, the Viterbi detector ***100*** is a time-varying PR4 detector, and the sync mark has the bit pattern given in Table I above.

[46] In addition to this embodiment of the Viterbi detector ***100*** being pruned like the **FIG. 7A** Viterbi detector, the sample clock is synchronized to the sync mark such

that the detector ***100*** is time varying. More specifically, referring to Table I, the logic 0's and 1's of the sync mark always come in pairs. Therefore, at every other sample time, the only possible states of the sync mark are S0 or S3. Consequently, by identifying the first sample of the sync mark and configuring the detector ***100*** such that this first sample is aligned with the sample time k+1 of the trellis, the detector "knows" that at k and k+2 only states S0 and S3 are possible. Therefore, one can eliminate all branches entering states S1 and S2 at sample times k and k+2. But because the trellis between k and k+1 differs from the trellis between k+1 and k+2, the detector ***100*** is said to be time varying because the trellis depends on the sample time. Even so, because there are only four branches between the states at each consecutive sample time, the time-varying Viterbi detector is often less complex and more robust than the non-time-varying Viterbi detector discussed above in conjunction with **FIG. 7A**.

[47] Furthermore, like the **FIG. 7A** Viterbi detector, this embodiment of the Viterbi detector ***100*** can recover the sync mark regardless of its polarity, and thus regardless of the head-connection polarity. Specifically, the trellis is symmetrical about the imaginary horizontal axis ***120*** between states S1 and S2. One may notice that because the sync mark of Table I has pairs and only pairs of logic 1's, the branches ***122*** and ***124*** can also be eliminated because the sync mark cannot have the state S3 at sample time k+1. But removing the branches ***122*** and ***124*** would destroy the symmetry about the imaginary axis ***120***, and would thus render the Viterbi detector ***100*** polarity dependent. That is, if the head-connection polarity were inverted, the detector ***100*** would be unable to recover the sync mark. Consequently, the servo circuit ***60*** would be unable to compensate for the inverted head-connection polarity.

[48] **FIG. 7C** is a two-sample-at-a-time version of the one-sample-at-a-time trellis diagram of **FIG. 7B**. Specifically, in this embodiment the sample circuit ***76***, the Viterbi detector ***78***, and the Viterbi detector ***100*** process two samples of the servo signal at a time. Therefore, the trellis of **FIG. 7C** is merely the trellis of **FIG. 7B** modified to reflect that the Viterbi detector ***100*** processes two samples at a time. Furthermore, the Viterbi detector ***100*** is non-time-varying when it processes two samples at a time.

[49] In one embodiment, the two-sample-at-a-time Viterbi detector ***100*** calculates a difference metric instead of path metrics, and updates the contents of the path history registers ***102*** based on the difference metric. Consequently, the Viterbi detector ***100*** can include circuitry that is less complex than would be needed if it calculated path metrics.

[50] The calculation of the difference metric is derived from the following PR4 path-metric equations, which use the following variables: PM00 equals the path metric for the state S0, PM11 equals the path metric for the state S1, Yf equals the first sample of a pair of samples (corresponds to k, k+2, k+4), Ys equals the second sample of a pair of samples (corresponds to k+1 and k+3, which are not shown in **FIG. 7C**), DM equals the difference metric = ½(PM00 – PM11), and Yk = Yf + Ys. As discussed above, each sample of a pair of samples has the same value. That is each pair of samples is either 00 or 11. Thus, the complexity of the of the Viterbi detector ***100*** is equivalent to the complexity of a single interleaved PR4 detector.

(4) If $PM00_k < PM11_k + (Yf + 1)^2 + (Ys + 1)^2$
Then $PM00_{k+1} = PM00_k$
Else $PM00_{k+1} = PM11_k + (Yf + 1)^2 + (Ys + 1)^2$

(5) If $PM11_k < PM00_k + (Yf - 1)^2 + (Ys - 1)^2$
Then $PM11_{k+1} = PM11_k$
Else $PM11_{k+1} = PM00_k + (Yf - 1)^2 + (Ys - 1)^2$

[51] Simplifying equations (4) and (5) to eliminate the square terms results in the following corresponding equations:

(6) If $PM00_k < PM11_k + 2Yf + 2Ys + 2$
Then $PM00_{k+1} = PM00_k$
Else $PM00_{k+1} = PM11_k + 2Yf + 2Ys + 2$

(7) If $PM11_k < PM00_k - 2Yf - 2Ys + 2$
Then $PM11_{k+1} = PM11_k$
Else $PM11_{k+1} = PM00_k - 2Yf - 2Ys + 2$

[52] Simplifying equations (6) and (7) by incorporating DM and Yk in the inequalities results in the corresponding equations:

(8) $Yk > DM_k - 1$
(9) $Yk < DM_k + 1$

[53] If equation (8) is false and equation (9) is true, then the Viterbi detector ***100*** updates DM and the path history registers PH00 and PH11 as follows, where 0 is the first (most recent) bit position and n is the last (least recent) bit position of the path registers:

(10) $DM_{k+1} = Yk + 1$
(11) $PH00\ (0{:}n)_{k+1} = [0, 0, PH11(0{:}n\text{-}2)_k]$
(12) $PH11(0{:}n)_{k+1} = [1, 1, PH11(0{:}n\text{-}2)_k]$

That is, the Viterbi detector ***100*** loads logic 0's into the two most recent bit positions 0 and 1 of PH00 and loads the remaining bit positions 2 - n with the contents of the corresponding bit positions 0 – n-2 of PH11. Next, the Viterbi detector ***100*** loads logic 1's into the two most recent bit positions 0 and 1 of PH11 while or after PH11 shifts the contents of its bit positions 0 – n-2 into its bit positions 2 – n.

[54] If equation (8) is true and equation (9) is false, then the Viterbi detector ***100*** updates DM and the path history registers PH00 and PH11 as follows:

(13) $DM_{k+1} = Yk - 1$
(14) $PH00(0{:}n)_{k+1} = [0, 0, PH00(0{:}n\text{-}2)_k]$
(15) $PH11(0{:}n)_{k+1} = [1, 1, PH00(0{:}n\text{-}2)_k]$

That is, the Viterbi detector ***100*** loads logic 1's into the two most recent bit positions 0 and 1 of PH11 and loads the remaining bit positions 2 - n with the contents of the corresponding bit positions 0 – n-2 of PH00. Next, the Viterbi detector ***100*** loads logic 0's into the two most recent bit positions 0 and 1 of PH00 while or after PH00 shifts the contents of its bit positions 0 – n-2 into its bit positions 2 – n.

[55] If both equations (8) and (9) are true, then the Viterbi detector ***100*** updates DM and the path history registers PH00 and PH11 as follows:

$$DM_{k+1} = DM_k \quad (16)$$

$$PH00(0:n)_{k+1} = [0, 0, PH00(0:n-2)_k] \quad (17)$$

$$PH11(0:n)_{k+1} = [1, 1, PH11(0:n-2)_k] \quad (18)$$

That is, when both equations (8) and (9) are true, the Viterbi detector ***100*** loads logic 0's into the two most recent bit positions 0 and 1 of PH00 while or after PH00 shifts the contents of its bit positions 0 – n-2 into its bit positions 2 – n. Similarly, the Viterbi detector ***100*** loads logic 1's into the two most recent bit positions 0 and 1 of PH11 while or after PH11 shifts the contents of its bit positions 0 – n-2 into its bit positions 2 – n.

[56] Equations (8) and (9) cannot both be false.

[57] **FIG. 8** is a table of Gray coded bit patterns ***130*** that form portions of the respective track identifiers ***42*** (**FIG. 3**) for eight adjacent tracks 0 – 7 (**FIG. 2**), and the corresponding uncoded bit patterns ***132*** according to an embodiment of the invention. The Gray coded bit patterns ***130*** include pairs and only pairs of consecutive logic 1's that are separated by no fewer than two consecutive logic 0's; therefore, the bit patterns ***130*** are compatible with the embodiments of the Viterbi detector ***100*** discussed above in conjunction with **FIGS. 5 – 7C**. Because the bit patterns ***130*** are compatible with the Viterbi detector ***100***, they allow the manufacturer to simplify the servo circuit ***60*** (**FIG. 5**) by eliminating the Viterbi detector ***78*** and using the Viterbi detector ***100*** to recover all of the servo data as discussed above in conjunction with **FIGS. 5** and **6**. The coding scheme used to generate the Gray coded bit patterns ***132*** is discussed in

commonly owned U.S. Patent Application Ser. No. _____ (Atty. Docket No. 01-S-045 (1678-39)) entitled "A DATA CODE AND METHOD FOR CODING DATA", which is incorporated by reference.

[58] **FIG. 9** is a block diagram of a disk-drive system ***200*** that incorporates the servo circuit ***60*** of **FIG. 5** according to an embodiment of the invention. The disk-drive system ***200*** includes a disk drive ***202***, which includes a read-write head ***204***, a write channel ***206*** for generating and driving the head ***204*** with a write signal, and a write controller ***208*** for interfacing the write data to the write channel ***206***. The head ***204*** may be similar to the head ***14*** of **FIG. 1**. The disk drive ***202*** also includes a read channel ***210***, which incorporates the servo circuit ***60*** **(FIG. 5)** for receiving a servo signal from the head ***204*** and for recovering servo data therefrom, and for providing the recovered servo data to a head-position circuit ***212***. The read channel ***210*** also receives an application-data read signal and recovers application data therefrom. The disk drive ***202*** also includes a read controller ***213*** for organizing the read data. Together, the write and read controllers ***208*** and ***213*** compose a disk-drive controller ***214***. The disk drive ***202*** further includes a storage medium such as one or more disks ***215***, each of which may contain data on one or both sides and which may be a magnetic, optical, or another type of storage disk. For example, the disks ***215*** may be similar to the disk ***12*** of **FIG. 1**. The head ***204*** writes/reads the data stored on the disks ***215***, and is connected to a movable support arm ***216***, which may be similar to the support arm ***16*** of **FIG. 1**. The head-position circuit ***212*** provides a control signal to a voice-coil motor (VCM) ***218***, which positionally maintains/radially moves the arm ***216*** so as to positionally maintain/radially move the head ***204*** over the desired data tracks on the disks ***215***. The VCM ***218*** may be similar to the VCM ***18*** of **FIG. 1**. A spindle motor (SPM) ***220*** and a SPM control circuit ***222*** respectively rotates the disks ***215*** and maintains them at the proper rotational speed.

[59] The disk-drive system ***200*** also includes write and read interface adapters ***224*** and ***226*** for respectively interfacing the disk-drive controller ***214*** to a system bus ***228***, which is specific to the system used. Typical system busses include ISA, PCI, S-Bus, Nu-Bus, etc. The system ***200*** typically has other devices, such as a

random access memory (RAM) ***230*** and a central processing unit (CPU) ***232*** coupled to the bus ***228***.

[60] From the foregoing it will be appreciated that, although specific embodiments of the invention have been described herein for purposes of illustration, various modifications may be made without deviating from the spirit and scope of the invention.